Proceedings of the 2007 IEEE/RSJ International
Conference on Intelligent Robots and Systems
San Diego, CA, USA, Oct 29 - Nov 2, 2007

A Scheduling and Routing Algorithm for Digital Microfluidic Ring Layouts with Bus-phase Addressing

Megha Gupta and Srinivas Akella

Abstract— **Digital microfluidic systems (DMFS) are a new class of lab-on-a-chip systems for biochemical analysis. A DMFS uses electrowetting to manipulate discrete droplets on a planar array of electrodes. The chemical analysis is performed by repeatedly moving, mixing, and splitting droplets on the electrodes. Recently, there has been a lot of interest in developing algorithms and computational tools for the design, simulation, and performance evaluation of DMFS. In this paper, we present an algorithm for coordinating droplet movement in batch mode operations on ring layouts with bus-phase addressing. In bus-phase systems, each electrode is not individually addressable, instead a set of electrodes are all controlled by the same signal. Though this hardware design simplifies chip fabrication, it increases the complexity of routing droplets. The presented algorithm allows multiple independent reactions, each with two reactants and one product, and chain reactions with multiple stages, where each stage produces reactants for the next stage, to take place simultaneously on the chip. This algorithm is scalable to different number of reactions within a limit which depends on the size of the layout, placement of sources and number of phases used. It also addresses any sensor constraints under which droplets need to visit sensor locations for specified amounts of time. We present simulation results using our algorithm to coordinate droplet movements for example analyses on a ring layout.**

I. INTRODUCTION

Lab-on-a-chip systems based on digital microfluidic technology have recently generated a lot of interest in the field of biochemical analysis. Digital microfluidic systems (DMFS) manipulate discrete nanoliter-sized droplets on a planar array of electrodes. These systems can be used for rapid automated biochemical analysis, thus impacting a wide variety of applications including biological research, genetic analysis, and biochemical sensing. The biochemical analysis is performed by repeatedly moving, mixing, and splitting droplets on the array. DMFS technology offers scalability, programmability, reconfigurability, and power reduction, and enables analysis of intermediate product droplets. These miniature systems drastically reduce the size of the equipment and the amounts of reagents used for the analysis. However, DMFS systems currently are programmed manually which greatly limits the scope of this technology, particularly when the number of droplets is large. Therefore, our goal is to design algorithms for the automated scheduling and routing of droplets in these systems.

This work was supported in part by NSF under Award No. IIS-0093233 and Award No. IIS-0541224.

The authors are with the Department of Computer Science, Rensselaer Polytechnic Institute, Troy, NY 12180 USA {guptam, sakella}@cs.rpi.edu

Our focus in this paper is on DMFS systems that consist of ring layouts with bus-phase addressing [1]. In *bus-phase systems*, each electrode is not directly addressable. Instead, a set of electrodes are controlled by the same signal and are said to have the same *phase*. This design makes fabrication easier and minimizes the number of electrical contacts. However, the complexity of droplet routing goes up significantly because a bus-phase system requires operations to be synchronized and imposes constraints on the parallelism achievable in the analysis. *Ring layouts with bus-phase addressing*, described in detail in Section 3, have been used to demonstrate droplet manipulation and pipelined glucose assays[1] [1]. We are not aware of algorithms for automated control of droplets in these systems.

We present an approach to completely automate batch mode operations on ring layouts. For batch mode, we take one droplet each of all reactants to produce one droplet each of all final products. Droplet routes are dynamically chosen and the same set of electrodes is shared among all droplets for transport, mixing and chemical reaction. Phase assignment for a layout can be done at the fabrication stage and once that is done, different reactions can be carried out on the same layout without changing the phases. The algorithm scales with the array size; it can handle a larger number of droplets on larger layouts.

II. RELATED WORK

Electrowetting [2], where the interfacial tension of droplets is modulated by a voltage, is an important method of actuation in a DMFS. A droplet moves to an adjacent electrode when the electrode is activated. Thus, by deactivating the electrode the droplet is present on, and activating the adjacent electrode simultaneously, a droplet can be manipulated on an array of electrodes as desired.

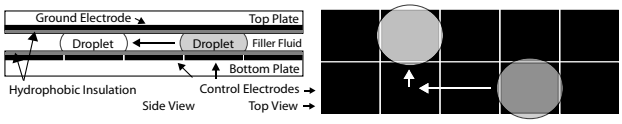

Fig. 1. Droplets on an electrowetting array (side and top views). The droplets are in a medium (usually oil or air) between two glass plates. The gray and white droplets represent the same droplet in initial and final positions. A droplet moves to a neighboring activated electrode which is turned off when the droplet has completed its motion. Based on [2].

Pollack, Fair, and Shenderov [2] first demonstrated this kind of manipulation of discrete microdroplets (Fig. 1). Fair

[1]Example videos of droplet manipulation on a ring layout may be seen at http://www.rsc.org/suppdata/lc/b4/b403341h/.

et al. [3] described experiments on dispensing, dilution, and mixing of samples in an electrowetting DMFS. Cho, Moon, and Kim [4] created, merged, split, and moved droplets using electrodes covered with dielectrics in an air environment. Gong, Fan, and Kim [5] developed a portable digital microfluidics lab-on-chip platform using electrowetting on dielectrics. Recent DMFS research has also focused on applications. Pollack et al. [6] demonstrated the use of electrowetting-based microfluidics for real-time polymerase chain reaction (PCR) applications. Srinivasan et al. [7] demonstrated the use of a DMFS as a biosensor for glucose, lactate, glutamate and pyruvate assays, and for clinical diagnostics on human blood, urine, saliva, sweat, and tears.

Ding, Chakrabarty, and Fair [8] described an architectural design and optimization methodology for scheduling biochemical analysis. They identified a basic set of droplet operations and used an integer linear programming formulation to minimize completion time. Here droplet paths and areas on the array for storage, mixing and splitting operations must be predefined by the user. Bohringer [9] pointed out that each droplet in a DMFS can be viewed as a simple robot that moves on a 4-connected array. He outlined an A* search approach for moving droplets from start to goal locations in prioritized order. However in his approach, the user must specify the locations for all droplet operations. Su and Chakrabarty [10] proposed architectural-level VLSI synthesis techniques for digital microfluidics-based biochips, and described integer programming and heuristic formulations to schedule assay operations, prior to location instantiation on the array. Griffith and Akella [11] developed layout designs and routing algorithms for arrays with direct addressing that can deal with a large number of droplets. Griffith, Akella, and Goldberg [12] presented polynomial-time algorithms for coordinating droplet movement on arrays with limited row-column addressing. Xu and Chakrabarty [13] suggested a droplet-trace-based method for array partitioning and phase assignment in a pin-constrained layout. Here the droplet routes must be known beforehand and separate areas on the array are reserved exclusively for different droplets. All these algorithms for automated control of droplets on DMFS platforms typically assume electrodes to be individually addressable; none are designed for bus-phase systems. Our algorithm completely automates droplet manipulation on ring layouts with bus-phase addressing.

III. RING LAYOUT WITH BUS-PHASE ADDRESSING

The algorithm has been designed for a particular type of DMFS layouts: ring layouts with bus-phase addressing [1]. The five components of a ring layout (Fig. 2) are as follows:

1) **Reservoir:** All sources from which reactants are introduced into the DMFS and all sinks to which waste and product droplets are sent, are classified as reservoirs. Each reservoir has a storage area and a neck. A droplet leaves or enters the storage area via the neck.
2) **Outer ring:** It is a planar ring of electrodes with a fixed number of phases allotted to its electrodes in a cyclic order. In Fig. 2, four phases (1,2,3,4) have been assigned to the outer ring. All reservoirs are directly connected to the outer ring.
3) **Inner ring:** It is a planar ring of electrodes that allows reactions to proceed to completion. A distinct set of phases is allotted to its electrodes in a cyclic order. It may have detection sites at some of its electrodes to sense the results of reactions. In Fig. 2, three phases (12,13,14) have been assigned to the inner ring.
4) **Mixer:** Any kind of mixing and splitting of droplets happens in the mixer. Mixing refers to the physical merging of two droplets that are supposed to react with each other and splitting refers to the formation of two product droplets from the merged droplet. The mixer is directly connected to both the rings. Seven distinct phases are required for the mixer. In Fig. 2, phases (5,6,7,8,9,10,11) have been assigned to the mixer.
5) **Interconnection:** It connects the outer ring and the inner ring and allows transport between them. Two distinct phases for each interconnection are required. In Fig. 2, phases (15,16) and (17,18) have been assigned to the left and right interconnections respectively.

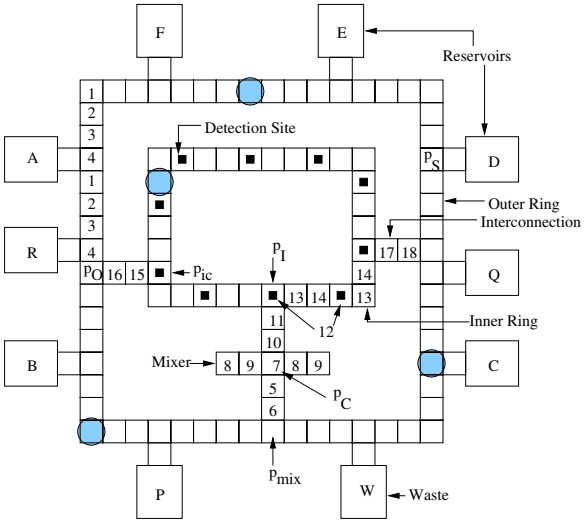

Fig. 2. Ring layout with bus-phase addressing: There are four phases for the outer ring, three for the inner ring, seven for the mixer and two for each of the interconnections. Within each ring, the phases are allocated among the electrodes in cyclic order. Thus, every fourth electrode in the outer ring has the same phase and every third electrode in the inner ring has the same phase. Based on [1].

IV. THE ALGORITHM

A. Problem Statement

Goal: Given a ring layout with bus-phase addressing, and a set of reactions, schedule and route the droplets so as to complete the chemical analysis, without any inadvertent mixing. Starting with one droplet of each reactant, a droplet of each final product must be produced with all timing constraints on mixing, reacting, and sensing satisfied.

Input: Number of electrodes in the rings, locations of mixer, inner ring, interconnections and reservoirs, reagents stored in

each reservoir, electrode phase assignments, reactions to be carried out, and mixing, reaction, and sensing times.
Output: Electrode activation schedule to coordinate droplet routing and to sense results of reactions.

We consider two classes of reactions: *independent* and *chain* reactions. The independent reactions form no intermediate products while the chain reactions consist of a series of reactions, each reaction defining a *stage*, whose products are used as reactants for the next stage. Each independent reaction and every stage of a chain reaction is assumed to consist of two reactants that give one product. An example set of independent reactions is:

$$A + B \rightarrow P$$
$$C + D \rightarrow Q$$
$$E + F \rightarrow R,$$

where A, B, C, D, E and F are chemical reagents to be mixed and P, Q, R are the final products. A possible assignment of reservoirs for this example is shown in Fig. 2. An example of a chain reaction (Fig. 3) is:

$$A + B \rightarrow C$$
$$C + D \rightarrow E$$
$$E + F \rightarrow G$$
$$G + H \rightarrow I,$$

where C, E, G are intermediate products and I is the final product. For the example above, stage 1 refers to the first reaction in the chain and stage 4 refers to the last reaction. We will call a pair of droplets of types that are supposed to react with each other a *compatible pair* of droplets. Thus, (A, B) form a compatible pair, and we have a total of four compatible pairs in the above example.

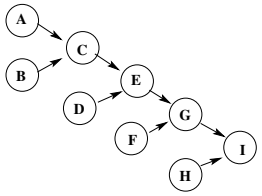

Fig. 3. Example of a chain reaction with four stages. Each stage produces one of the reactants for the next stage. C, E and G are intermediate products while I is the final product.

B. Assumptions

1) The mixer is used only for physical mixing of droplets. It can be entered from both the rings for chain reactions but only from the outer ring for independent reactions.
2) Any new droplet formed in the mixer that is neither a final product nor an intermediate product is labeled a waste droplet. Such droplets exit only to the outer ring from the mixer and are directed to the waste outlet.
3) The inner ring is used for completion of chemical reaction and for sensing of droplets. It can be entered only from the mixer.
4) In both the rings, droplets move in the anticlockwise direction only and the rings may be deactivated, i.e., paused completely, when desired. This means that none of the phases in a ring are activated, thus stopping all droplet motion in that ring. When they are reactivated, they resume from current positions.

C. Overview

The sequence of steps to be carried out for any reaction is essentially the same: 1) Introduce the droplets into the system from sources. 2) Merge compatible pairs of droplets in the mixer. Move the merged droplet multiple times in the mixer for the input mixing time, t_{mix}, to ensure uniformity in composition inside the droplet. 3) Split the mixed droplet to form two regular-sized product droplets. Send one of them to the inner ring for completion of the chemical reaction and the other to the waste outlet. 4) Retain the product droplet in the inner ring for the prescribed reaction time, t_{react}, to allow the chemical reaction to complete. 5) For any sensing requirements, sense the product droplet for the input sensing time, t_{sense}. 6) Send final products to their respective sinks.

The algorithm generates proper electrode activations for all the above steps such that there are no unwanted droplet movements or mixing throughout the analysis, and all timing constraints are satisfied. Reagent transport and droplet mixing occur on the same shared set of electrodes. Since multiple electrodes are controlled by the same signal, activation of an electrode with a particular phase can result in droplet movements at all other electrodes with the same phase. Hence, the complexity of preventing undesired mixing between droplets is greater for bus-phase systems.

Let the current position of droplet i be denoted by p_i and that of an electrode e by p_e. Specifically, let us denote entry to the mixer by p_{mix}, the central electrode of the mixer by p_C, an electrode adjacent to any source or sink by p_S, entry to the inner ring by p_I, and any entry to the outer ring from the inner ring by p_O. We will consider the anticlockwise direction as positive. So, p_{e-n} refers to the electrode that is n electrodes before the electrode e. All current droplet positions are calculated and updated by the algorithm after every clock cycle so that the droplet positions are always known globally. We will now describe in detail how each of the operations like droplet dispensing, routing, mixing, and splitting is handled by the algorithm.

D. Sources

For chain reactions, we assume the introduced droplets are at the right concentration. Before the droplet is introduced into the outer ring, the source electrodes are ON, neck electrode is OFF and p_S may be ON or OFF. To introduce the droplet, use Algorithm 1. The sources may or may not be connected to different phases of the outer ring. If the neck electrodes of two sources, S and S', connected to the same phase i of the outer ring have the same phase j, then to introduce a droplet from S, j will have to be activated followed by i. This will lead to a droplet from S' also getting introduced into the outer ring even though it was not planned. To avoid this, the neck electrode of each source should have a unique phase.

Algorithm 1: Droplet injection from sources

1 **if** p_{S-2} *is empty when* p_{S-2} *is activated* **then**
2 Activate neck electrode and turn off source electrodes simultaneously when p_{S-1} turns on.
3 Switch off neck electrode when p_S turns on so that one droplet is introduced into the outer ring.
4 Activate source electrodes again.
5 **else** Go to step 1.

E. Mixing

We employ a greedy algorithm to introduce droplets into the mixer. Though this may not give us a globally optimum solution in all cases, it is simple to implement and significantly reduces computational overhead. The global optimum depends on a number of factors like the mixing times and time taken by the split droplets to leave the mixer. The second factor in turn depends on availability of space in the outer and inner rings, which changes dynamically. We will now describe the greedy scheduling algorithm for two cases.

Case 1: Mixer is empty and there is at least one compatible pair of droplets in the outer ring. At every clock cycle, compute $max\{d(p_X, p_{mix}), d(p_Y, p_{mix})\}$ for every compatible pair (X, Y) in the outer ring, where p_X and p_Y represent positions of droplets X and Y and $d(u, v)$ = 1 + number of electrodes between the two positions u and v. This expression calculates for every compatible pair of droplets, the distance of the droplet which is farther away from the mixer. The pair of droplets for which this distance is the least can be brought to the mixer in the least number of clock cycles. Therefore, choose that pair of droplets to be sent to the mixer. Suppose A and B are chosen. Then, the minimum number of clock cycles required is $max(d(p_A, p_{mix}), d(p_B, p_{mix}))$.

When a droplet is at p_{mix}, pause the outer ring and activate phase 6. Once a droplet has entered the mixer, the outer ring can be restarted and simultaneously, phase 5 of the mixer is activated followed by phase 7 so as to bring the droplet to p_C. This will have to be done twice, first at clock cycle $1 + min(d(p_A, p_{mix}), d(p_B, p_{mix}))$ when the first droplet arrives at p_{mix} and then $1 + d(p_A, p_B)$ clock cycles later for the second droplet. If another droplet is already present at p_C, phase 10 is activated before phase 5.

In the case of chain reactions, the intermediate droplets can be brought into the mixer from the inner ring itself, once they have reacted completely according to prescribed reaction times. For this, when the droplet is at p_I, pause the inner ring and activate phase 11. Once the droplet has entered the mixer, the inner ring can be restarted and simultaneously, phase 10 of the mixer is activated followed by phase 7 so as to bring the droplet to p_C. If another droplet is already present at p_C, phase 5 is activated before phase 10.

A droplet's motion is affected by interference by other droplet motions and electrode activations. For example, the outer ring may need to be paused to allow a droplet to enter a sink. So, the number of actual clock cycles may be more than the number of theoretical minimum clock cycles calculated.

Case 2: Mixer is empty but no compatible pair of droplets is present in the outer ring. In this case, do not let any of the droplets enter the mixer till at least one compatible pair of droplets is found in the outer ring and then, we have obtained an instance of Case 1 described above.

Once both types of droplets are present in the mixer, we look up the required physical mixing time, t_{mix}, for the pair and repeat the following sequence of electrode activations for t_{mix} clock cycles to enable mixing to occur: $8 \rightarrow 9 \rightarrow 8 \rightarrow 7 \rightarrow 9 \rightarrow 8 \rightarrow 9 \rightarrow 7$. If at the end of t_{mix} cycles, the merged droplet is not at p_C, continue with the above sequence of activations till the droplet reaches p_C.

It is important to note that the above algorithm can be carried out exactly the same way for a ring layout with multiple mixers too because the algorithm depends only on p_{mix}, p_I, and phases of the mixer. Therefore, each mixer can use the above algorithm with its own values of p_{mix}, p_I, and phases.

F. Splitting

The merged droplet is double the volume of regular droplets. It can be split by activating the two electrodes on its either side simultaneously, i.e., by turning off phase 7 and activating phases 5 and 10 simultaneously. Splitting results in two regular-sized droplets, one on phase 5 and the other on phase 10. Note that if p_I is at phase j, we can introduce a droplet at p_I only when phase j is activated. So, if a droplet is to be sent to the inner ring, it is first checked if p_{I-2} is empty when it is activated because if there is already a droplet at p_{I-2}, it would reach p_I when phase j is activated and mix with the droplet from the mixer. If both split droplets have to be sent to the inner ring, use Algorithm 2. If one of the split droplets has to be sent to the waste outlet via the outer ring, then use Algorithm 3. Again, these procedures can be used for a ring layout with multiple mixers and multiple inner rings because they depend only on p_{mix}, p_I, and phases of the mixer. Therefore, each mixer and inner ring will need to use a distinct set of phases and then can use these procedures with its own values of p_{mix}, p_I, and phases.

G. Chemical Reaction Completion and Sensor Constraints

If some sensor constraints have been specified by the user, optical detection sites are located in the inner ring to fulfill them. For example, the droplet may be required to stay on a detection site for a prescribed number of clock cycles, say t_{sense}, after the completion of the chemical reaction. In Fig. 2, every inner ring electrode with phase 12 is a detection site. Algorithm 4 allows the reactions to complete and takes the sensor constraints into account accordingly.

H. Interconnections

Each interconnection requires two distinct phases. If the same pair of phases is used for two interconnections $IC1$ and $IC2$, then unwanted droplet movement can occur around $IC2$ when any phase of $IC1$ is activated. Suppose phases 15 and 16 are used for one of the interconnections as shown in Fig. 2. Let us call the inner ring electrode adjacent to

the interconnection p_{ic}. Algorithm 5 allows movement of droplets from the inner to the outer ring.

Algorithm 2: Movement of split droplets out of mixer(a)

```
1 Split the merged droplet.
2 while the mixer is not empty do
3     if p_{I-2} is empty when it is activated then
4         Activate phase 11 when p_{I-1} is activated so
          that one of the split droplets moves toward the
          inner ring.
5         Activate phase 7 in the next clock cycle so that
          the other droplet reaches p_C.
6         Activate phase 10 and switch off phase 11 when
          phase j is activated so that the first droplet
          enters the inner ring while the second one
          moves toward it.
7     end
8 end
```

Algorithm 3: Movement of split droplets out of mixer(b)

```
1 Split the merged droplet. Perform steps 2 and 5 below
  simultaneously.
2 if p_{I-2} is empty when it is activated then
3     Activate phase 11 when p_{I-1} is activated so that
      one of the split droplets moves toward the inner
      ring. This droplet would now enter the inner ring
      when phase j is activated.
4 else Go to step 2.
5 if p_{mix-2} is empty when it is activated then
6     Activate phase 6 when p_{mix-1} is activated so that
      the waste droplet moves toward the outer ring. This
      droplet would now enter the outer ring when
      electrode p_{mix} is activated.
7 else Go to step 5.
```

Algorithm 4: Chemical reaction and droplet sensing

```
1  Move a droplet in the inner ring for t_{react} clock cycles
   so that the chemical reaction is completed.
2  if no sensor constraints specified then Go to step 9.
3  else
4      if the droplet is not on a detection site then
5          move it to the closest detection site.
6      Pause the inner ring for t_{sense} clock cycles.
7      Restart the inner ring.
8  end
9  if the droplet is a final product then
10     Calculate the location of the interconnection closest
       to the corresponding sink and set it as the droplet's
       destination.
11 else Set p_I as the droplet's destination to resend it to
   the mixer.
```

I. Sinks

As for sources, multiple sinks may be connected to the same phase of the outer ring and so, the neck electrode of each sink should have a unique phase. To send a droplet to sink $S1$, mark p_{S1} as the droplet's destination when it enters the outer ring from an interconnection and use Algorithm 6.

Algorithm 5: Movement from the inner to the outer ring

```
1  foreach droplet reaching p_{ic} do
2      if its destination is p_{ic} then
3          Pause inner ring.
4          if p_{O-3} is empty when it is activated then
5              Activate phase 15 and restart the inner ring
               when electrode p_{O-2} is turned on.
6              Switch off phase 15 and activate 16 when
               electrode p_{O-1} is on.
7              Turn off 16 when electrode p_O is activated
               so that the droplet moves to the outer ring.
8          end
9      end
10 endfch
```

Algorithm 6: Droplet removal from DMFS array

```
1 foreach droplet reaching p_{S1} do
2     if its destination is p_{S1} then
3         Pause outer ring and activate the neck electrode
          of S1 so that the droplet enters S1.
4         Turn off neck electrode and switch on sink
          electrodes. Then, restart the outer ring.
5     end
6 endfch
```

V. OPTIMALITY ANALYSIS OF THE GREEDY ALGORITHM

To better understand the performance of our greedy algorithm for scheduling mixing operations, we present a preliminary analysis for two reactions. Consider two compatible droplet pairs, $(A1, A2)$ and $(B1, B2)$. Let $A1$ and $B1$ be closer to p_{mix} than $A2$ and $B2$ respectively. For droplet i, let $d_i = d(p_i, p_{mix})$. Without loss of generality, let $d_{A2} < d_{B2}$. This can happen in three ways as shown in Fig. 4.

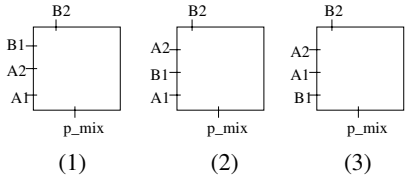

Fig. 4. The three possibilities when the greedy algorithm will schedule the droplet pair $(A1, A2)$ before $(B1, B2)$, i.e., when $d_{A2} < d_{B2}$.

$d_{A2} < d_{B2}$ means the greedy algorithm, G, will always schedule $(A1, A2)$ before $(B1, B2)$. Let OPT' be a schedule that always schedules $(B1, B2)$ before $(A1, A2)$. Let the number of clock cycles required by G to complete mixing of both pairs be T_G and that required by OPT' be $T_{opt'}$. We will now analyze whether G is optimal for the three cases in Fig. 4 with zero and finite mixing times.

1) t_{mix} = 0 for both pairs.
   Case 1: $d_{A2} < d_{B1}$. Here, G takes d_{A2} cycles to take $(A1, A2)$ in the mixer. By then, $B2$ has moved d_{A2} electrodes closer to p_{mix}. So, G takes another $(d_{B2} - d_{A2})$ cycles to take $(B1, B2)$ in the mixer. On the other

hand, OPT' takes d_{B2} cycles to take $(B1, B2)$ in the mixer. By then, $A2$ has moved $(d_{B2} - d_{A2})$ electrodes past p_{mix}. So, G takes another $n_O - (d_{B2} - d_{A2})$ cycles to take $(A1, A2)$ in the mixer. Therefore,

$$\begin{aligned} T_G &= d_{A2} + (d_{B2} - d_{A2}) = d_{B2} \\ T_{opt'} &= d_{B2} + n_O - (d_{B2} - d_{A2}) = n_O + d_{A2} \end{aligned}$$

Case 2: $d_{A2} > d_{B1}$.

$$\begin{aligned} T_G &= d_{A2} + n_O - (d_{A2} - d_{B1}) = n_O + d_{B1} \\ T_{opt'} &= d_{B2} + n_O - (d_{B2} - d_{A2}) = n_O + d_{A2} \end{aligned}$$

Case 3: $d_{A1} > d_{B1}$.

$$\begin{aligned} T_G &= d_{A2} + n_O - (d_{A2} - d_{B1}) = n_O + d_{B1} \\ T_{opt'} &= d_{B2} + n_O - (d_{B2} - d_{A2}) = n_O + d_{A2} \end{aligned}$$

Thus, $T_G < T_{opt'}$ whenever t_{mix} = 0 for each pair. As there are only two possible schedules for two droplet pairs out of which G is better, it is also the optimal.

2) $t_{mix} > 0$ for both pairs.
Here, let us assume that t_{mix} for each pair is sufficiently large so that while it is mixing, the other pair goes past the mixer exactly once. Let t_{mix1} and t_{mix2} be the mixing times for the two pairs respectively.
Case 1: $d_{A2} < d_{B1}$.

$$\begin{aligned} T_G &= d_{A2} + t_{mix1} + n_O - (t_{mix1} - (d_{B2} - d_{A2})) \\ &\quad + t_{mix2} = n_O + d_{B2} + t_{mix2} \\ T_{opt'} &= d_{B2} + t_{mix2} + n_O - (t_{mix2} + d_{B2} - d_{A2}) \\ &\quad + t_{mix1} = n_O + d_{A2} + t_{mix1} \end{aligned}$$

Case 2: $d_{A2} > d_{B1}$.

$$\begin{aligned} T_G &= d_{A2} + t_{mix1} + n_O - (t_{mix1} - (d_{B2} - d_{A2})) \\ &\quad + t_{mix2} = n_O + d_{B2} + t_{mix2} \\ T_{opt'} &= d_{B2} + t_{mix2} + n_O - (t_{mix2} + d_{B2} - d_{A2}) \\ &\quad + t_{mix1} = n_O + d_{A2} + t_{mix1} \end{aligned}$$

Case 3: $d_{A1} > d_{B1}$.

$$\begin{aligned} T_G &= d_{A2} + t_{mix1} + n_O - (t_{mix1} - (d_{B2} - d_{A2})) \\ &\quad + t_{mix2} = n_O + d_{B2} + t_{mix2} \\ T_{opt'} &= d_{B2} + t_{mix2} + n_O - (t_{mix2} + d_{B2} - d_{A2}) \\ &\quad + t_{mix1} = n_O + d_{A2} + t_{mix1} \end{aligned}$$

Hence, in this case, whether $T_G < T_{opt'}$ or not depends on the distance of each compatible pair from the mixer and t_{mix} for each pair. So, G may not always be the best choice.

VI. SYSTEM DEADLOCK

System deadlock means that the system gets stuck in a particular state, preventing any further progress of the biochemical analysis. Let e_o and e_i be the number of electrodes in the outer and the inner ring respectively, n_i and n_{mix} the number of droplets in the inner ring and mixer respectively, w_{mix} the number of waste droplets in the mixer, and w_o the number of waste or final product droplets in the outer ring.

The key observation in calculating the number of droplets that can cause a deadlock is that the droplets can leave the system only through the outer ring. Therefore, if the outer ring is filled to its capacity but has no waste or final product droplets, and if the mixer has at least one waste droplet that cannot exit to the outer ring, the system reaches a deadlock as none of the droplets from the outer ring can leave the system or enter the mixer. Since the outer ring has four phases, every fourth electrode will have a droplet on it. Thus, the number of droplets in such a case is $(e_o/4 + n_i + n_{mix})$ where

$$0 \leq n_i \leq e_i/3; \quad 0 < n_{mix} \leq 2; \quad w_o = 0; \quad w_{mix} > 0.$$

A deadlock can also occur if both the split droplets must go to the inner ring. In this case, if both the rings are filled to their capacity and the outer ring has no waste or final product droplets, no movement from the inner ring to the outer ring or from the outer ring to the mixer can occur. Since the inner ring has three phases, every third electrode of the inner ring will have a droplet on it. Therefore, the number of droplets in this case is $(e_o/4 + e_i/3 + n_{mix})$ where

$$0 < n_{mix} \leq 2; \quad w_o = 0; \quad w_{mix} = 0.$$

In Fig. 2 , e_o = 60, e_i = 30, and we treat one of the split droplets as waste. Thus, as few as 16 droplets can cause a deadlock with 15 droplets in the outer ring, w_o = 0, and a waste droplet in the mixer.

VII. RESULTS

The algorithm has been implemented in C++ and simulated successfully for both independent and chain reactions. We define *completion time* of the analysis as the number of clock cycles required to reach the state when all reactions have been completed, and all product droplets on the array have been sent to the respective sinks or waste outlet, thus leaving no droplet on the array. For chain reactions, the algorithm has been optimized by defining a stage with each reaction, i.e., the stage at which each reaction in the chain occurs is given as an input. By doing so, the algorithm knows at all times the pair of droplets to be mixed next and this reduces the total transportation time of droplets. For example, at stage 2 of the chain reaction in Fig. 3, it is known that (C, D) is the pair to be sent to the mixer. So, D will be taken in by the empty mixer even if C is still in the inner ring waiting for the reaction to complete. Thus, the overall transportation time of D and the time for which the mixer is empty is reduced. For a test reaction with seven stages in the chain, t_{mix} = 5, t_{react} = 20, and t_{sense} = 5, the algorithm took 513 clock cycles to reach completion without any stages defined and 460 clock cycles with stages defined with each reaction, that is a reduction of 10% in completion time.

Another optimization for chain reactions was obtained by observing that an intermediate product droplet is in the inner ring for completion of chemical reaction and needs to be sent to the mixer for the next stage. So, its transportation time can be reduced by allowing droplets to enter the mixer from the inner ring itself. For the seven stage chain reaction, the algorithm took 460 clock cycles to reach completion without

this optimization and 387 cycles with the optimization, that is an additional 15% reduction in completion time.

We have simulated example independent and chain reactions operating in batch mode. Animation of a chain reaction with seven stages and sensor constraints has been attached with the paper. In the video, droplets having the same color represent a compatible pair. One of the droplets from each split operation is treated as waste. The gray reservoir is for waste droplets and the dark green reservoir is for the final product. Other animations of independent reactions, chain reactions without sensor constraints, and chain reaction with and without predefined stages can be seen at `http://www.cs.rpi.edu/~sakella/BusedDMFS`.

As discussed earlier, the algorithm extends to modified ring layouts with multiple mixers and multiple inner rings as well. This can be important as additional mixers can reduce the completion time for independent reactions and for chain reactions, multiple executions of the chain can be done in parallel to obtain multiple droplets of the final product in lesser time. We have also performed branched chain reactions, where multiple chains merge to form a single final product, by treating each stage as an independent reaction. An example of a branched chain reaction is the polymerase chain reaction (PCR) for DNA sequence analysis [6] (Fig. 5).

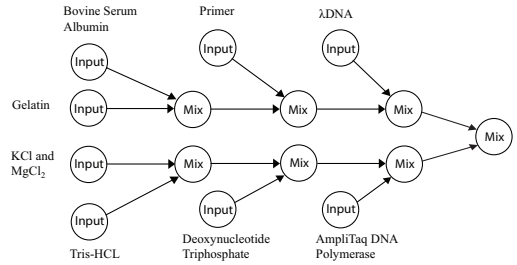

Fig. 5. PCR analysis graph.

VIII. CONCLUSIONS AND FUTURE WORK

We have presented an approach to carry out batch-mode operations on ring layouts with bus-phase addressing. We have successfully demonstrated automated independent and chain reactions in simulation and identified possible deadlock scenarios. The algorithm allows a variety of chemical analyses to occur in parallel on a given ring layout and also works for layouts with multiple mixers and inner rings, different numbers of reservoirs, and different phase assignments.

We plan to optimize the current algorithm by relaxing many of the simplifying assumptions. For example, droplet movement from the outer ring to the inner ring can be permitted to reduce transportation time to the mixer. Waste droplets can be allowed to leave the mixer via the inner ring. Both clockwise and anticlockwise movement can be permitted in both the rings. For a reaction with no sensor constraints, the droplet undergoing a reaction can be transported to its destination while the reaction goes to completion. Mixing times and reaction times can be taken into account while choosing the droplet pair to be sent to the mixer to get a more efficient scheduling strategy. We can also use the algorithm to evaluate performance with different phase assignments to assist chip design.

Changes to the ring layout such as multiple instances of the ring layout connected together, and use of multiple mixers and inner rings, can permit a greater number of reactions to take place on the array. Automating continuous mode reactions is also an important goal. Optimum input rate and placement of components need to be identified to avoid congestion and deadlocks. Finding an optimum schedule to minimize completion time and generalizing the algorithm to control arbitrary layouts with bus-phase addressing pose very interesting challenges.

ACKNOWLEDGMENTS

The authors are thankful to Nilanjan Chakraborty for many useful insights into the paper, and to Eric Griffith for early conversations on this topic.

REFERENCES

[1] V. Srinivasan, V. K. Pamula, and R. B. Fair, "An integrated digital microfluidic lab-on-a-chip for clinical diagnostics on human physiological fluids," *Lab on a Chip*, vol. 4, pp. 310–315, 2004.

[2] M. G. Pollack, R. B. Fair, and A. D. Shenderov, "Electrowetting-based actuation of liquid droplets for microfluidic applications," *Applied Physics Letters*, vol. 77, pp. 1725–1726, 2000.

[3] R. B. Fair, V. Srinivasan, H. Ren, P. Paik, V. Pamula, and M. G. Pollack, "Electrowetting-based on-chip sample processing for integrated microfluidics," in *IEEE International Electron Devices Meeting (IEDM)*, 2003, pp. 779–782.

[4] S. K. Cho, H. Moon, and C.-J. Kim, "Creating, transporting, cutting, and merging liquid droplets by electrowetting-based actuation for digital microfluidic circuits," *Journal of Microelectromechanical Systems*, vol. 12, no. 1, pp. 70–80, Feb. 2003.

[5] J. Gong, S.-K. Fan, and C.-J. Kim, "Portable digital microfluidics platform with active but disposable lab-on-chip," in *Tech. Digest of 17th IEEE International Conference on Micro Electro Mechanical Systems*, Maastricht, The Netherlands, Jan. 2004, pp. 355–358.

[6] M. G. Pollack, P. Y. Paik, A. D. Shenderov, V. K. Pamula, F. S. Dietrich, and R. B. Fair, "Investigation of electrowetting-based microfluidics for real-time PCR applications," in *Seventh International Conference on Miniaturized Chemical and Biochemical Analysis Systems (MicroTAS '03)*, Squaw Valley, CA, Oct. 2003.

[7] V. Srinivasan, V. Pamula, M. Pollack, and R. Fair, "A digital microfluidic biosensor for multianalyte detection," in *IEEE 16th Annual International Conference on Micro Electro Mechanical Systems*, 2003, pp. 327–330.

[8] J. Ding, K. Chakrabarty, and R. B. Fair, "Scheduling of microfluidic operations for reconfigurable two-dimensional electrowetting arrays," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 20, no. 12, pp. 1463–1468, Dec. 2001.

[9] K. F. Böhringer, "Modeling and controlling parallel tasks in droplet-based microfluidic systems," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 25, pp. 334–344, Feb. 2006.

[10] F. Su and K. Chakrabarty, "Architectural-level synthesis of digital microfluidics-based biochips," in *Proc. IEEE International Conference on CAD*, 2004, pp. 223–228.

[11] E. J. Griffith and S. Akella, "Coordinating multiple droplets in planar array digital microfluidic systems," *International Journal of Robotics Research*, vol. 24, no. 11, pp. 933–949, Nov. 2005.

[12] E. J. Griffith, S. Akella, and M. K. Goldberg, "Performance characterization of a reconfigurable planar-array digital microfluidic system," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 25, no. 2, pp. 340–352, Feb. 2006.

[13] T. Xu and K. Chakrabarty, "Droplet-trace-based array partitioning and a pin assignment algorithm for the automated design of digital microfluidic biochips," in *Proc. IEEE/ACM International Conference on Hardware/Software Codesign and System Synthesis*, 2006, pp. 112–117.